Corso di
Complementi di gasdinamica

Tommaso Astarita
astarita@unina.it
www.docenti.unina.it

Soluzione delle equazioni

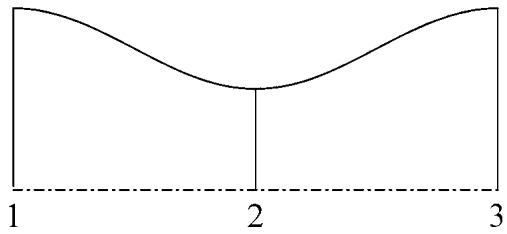
In generale la soluzione delle equazioni è abbastanza complessa ed è quindi necessario risolvere un'equazione differenziale alle derivate ordinarie (ODE) numericamente.

Il problema non presenta particolari difficoltà e per questo motivo si utilizzerà un metodo di Runge - Kutta del quarto ordine:

Soluzione delle equazioni

Definiamo una geometria standard di tipo sinusoidale:

$$\begin{cases} x < x_2 & A(x) = A_1 + (A_1 - A_2)\left(\cos\left(\pi \dfrac{x - x_1}{x_2 - x_1}\right) - 1\right)\dfrac{1}{2} \\ x < x_3 & A(x) = A_2 + (A_2 - A_3)\left(\cos\left(\pi \dfrac{x - x_2}{x_3 - x_2}\right) - 1\right)\dfrac{1}{2} \end{cases}$$

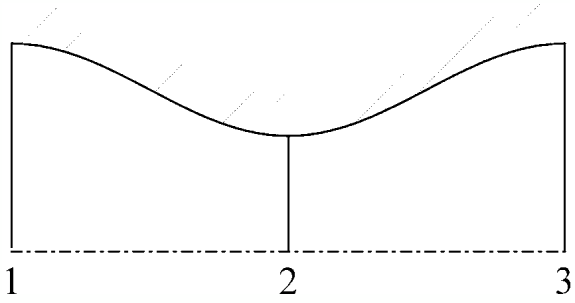



```
% \Moto1D_1\A.m
function A=A(x,Xs,AX)
cc=find(x<=Xs(2));
cc1=find(x>Xs(2));
A(cc)=AX(1)+(AX(1)-AX(2))*(cos(pi*(x(cc)-Xs(1))/(Xs(2)-Xs(1)))-1)/2;
A(cc1)=AX(2)+(AX(2)-AX(3))*(cos(pi*(x(cc1)-Xs(2))/(Xs(3)-Xs(2)))-1)/2;
```


Soluzione delle equazioni

Derivando si ha:

$$\begin{cases} x < x_2 & \dfrac{dA}{dx} = -\pi \dfrac{A_1 - A_2}{2(x_2 - x_1)} \sin\left(\pi \dfrac{x - x_1}{x_2 - x_1}\right) \\ x < x_3 & \dfrac{dA}{dx} = -\pi \dfrac{A_2 - A_3}{2(x_2 - x_1)} \sin\left(\pi \dfrac{x - x_2}{x_3 - x_2}\right) \end{cases}$$

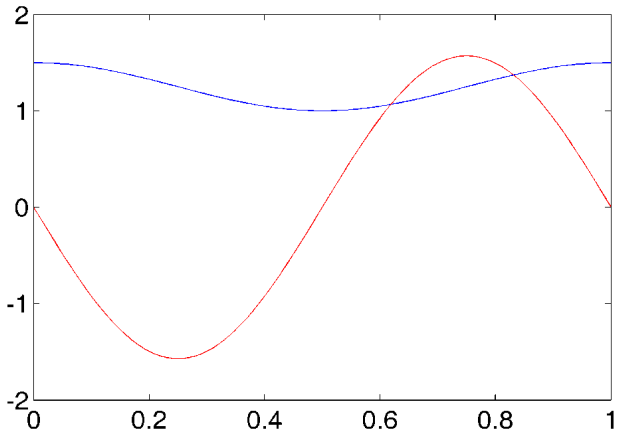



```
% \Moto1D_1\dAdx.m
function dAdx=dAdx(x,Xs,AX)
cc=find(x <= Xs(2) );
cc1=find(x > Xs(2) );
dAdx(cc)  = -pi* 0.5*(AX(1)- AX(2)) * sin(pi*(x(cc)  - Xs(1)) /(Xs(2) - Xs
(1)))/(Xs(2) - Xs(1));
dAdx(cc1) = -pi* 0.5*(AX(2)- AX(3)) * sin(pi*(x(cc1) - Xs(2)) /(Xs(3) -
Xs(2)))/(Xs(3) - Xs(2));
```

Soluzione delle equazioni

Diagrammando:

```
% \Moto1D_1\ProvaArea.m
PosA=[0;0.5;1];
DimA=[1.5;1;1.5];
x=0:.01:1;
clf
plot(x,A(x,PosA,DimA));
hold on
plot(x,dAdx(x,PosA,DimA),'r');
hold off
```



Soluzione delle equazioni

La seguente equazione differenziale può essere codificata come

$$(12)\quad \frac{1-M^2}{1+\frac{\gamma-1}{2}M^2}\frac{dM^2}{M^2} = -2\frac{dA}{A} + \gamma M^2 dN + \left(1+\gamma M^2\right)\frac{dT_o}{T_o} + 2\left(1+\gamma M^2\right)\frac{d\dot{m}}{\dot{m}}$$

```
%\Moto1D_1\dMdx.m
function dMdx = dMdx(x,y)
global PosA DimA gamma;
M2=y(1);%Mach al quadrato
Appo=(1-M2)/(1+M2*(gamma-1)/2);
if Appo==0 error ( 'Divisione per zero');end
Appo=M2/Appo;
Appo1=1+gamma*M2;
dAdxA=dAdx(x,PosA,DimA)/A(x,PosA,DimA);
dNdx=0;     %Funzione nota
dT0dxT0=0;  %Funzione nota
dmdxm=0;    %Funzione nota
dMdx=[Appo*(-2*dAdxA+gamma*M2*dNdx+Appo1*dT0dxT0+
2*Appo1*dmdxm)
    dAdx(x,PosA,DimA)];
```

Soluzione delle equazioni

Il programma principale diventa:

```
%\Moto1D_1\Moto1D.m
global PosA DimA gamma;
% quantita' note
PosA=[0;0.5;1];
DimA=[1.5;1;3];
gamma=1.4;

L=[0 0.5]; %dominio d'integrazione
M=.43;     %Mach iniziale
M0=[M^2 DimA(1)]; %Condizioni iniziali
h=0.005; %Passo d'integrazione
N=floor((L(2)-L(1))/h)+1;   %Numero di punti
x=zeros(N,1);    % Allocazione memoria
M=zeros(N,length(M0));% Allocazione memoria
```


Soluzione delle equazioni

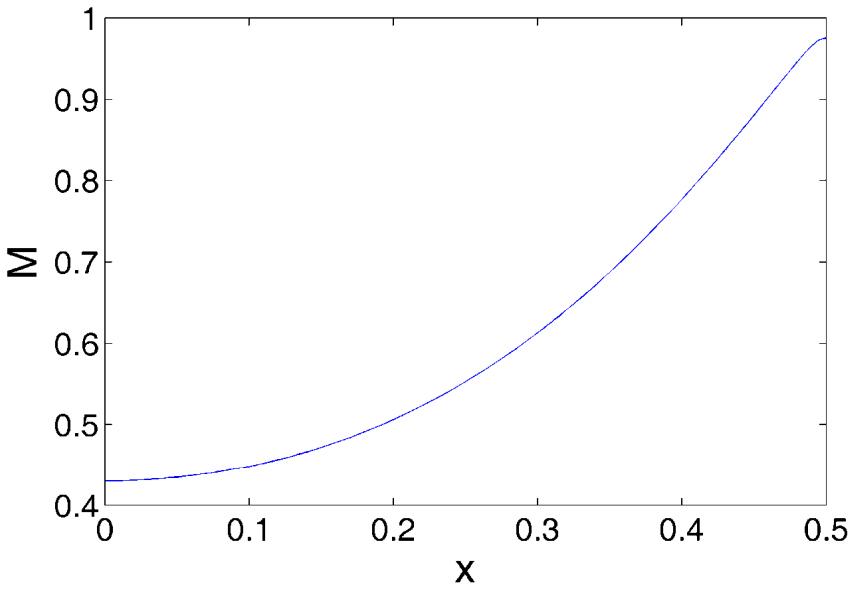
Il programma principale diventa:

```
x(1)=L(1);  %Condizioni iniziali
M(1,:)=M0;  %Condizioni iniziali
tic
for i=1:N-1
    [x(i+1) , M(i+1,:)]=rk4(x(i),h,M(i,:)',@dMdx);
end
toc
plot(x,sqrt(M(:,1)));
xlabel('x','FontSize',20);
ylabel('M','FontSize',20);
set(gca,'FontSize',16);
fprintf('M=%g\n'  ,sqrt(M(end,1)));
axis([0 0.5 0.4 1])
```

Soluzione delle equazioni

Il risultato è:

M=0.976208



Mentre utilizzando le tabelle si ha:

Dati		Sezione 2	
M	0.43	A2/A*	1.0004783
A1/A2	1.5	M2	0.9762034
A1/A*	1.500717501		

Soluzione delle equazioni

Utilizzando ode45 il programma principale diventa:

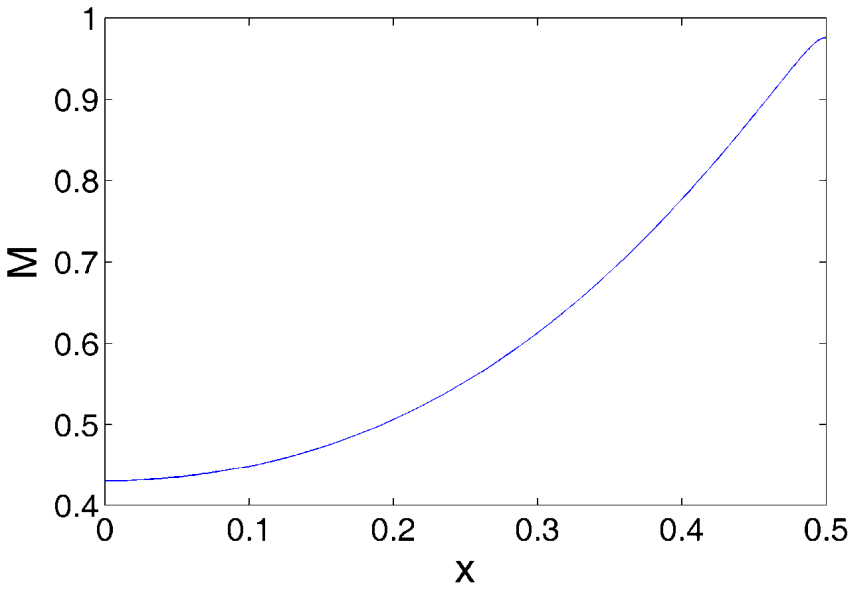
```
%\Moto1D_2\Moto1D.m
global PosA DimA gamma;
PosA=[0;0.5;1];  % quantita' note
DimA=[1.5;1;3];
gamma=1.4;
L= [0 0.5];% dominio d'integrazione
M=.43;
M0=[M^2 DimA(1)];  %Condizione iniziale

options = odeset('RelTol',1e-8, 'AbsTol',[1e-8]);
[x, M]=ode45(@dMdx,L,M0,options);
plot(x,sqrt(M(:,1)),'-' );
xlabel('x','FontSize',20);
ylabel('M','FontSize',20);
set(gca,'FontSize',16);
fprintf('M=%g\n' ,sqrt(M(end,1)));
```

Soluzione delle equazioni

Il risultato è:

M=0.976203



Mentre utilizzando le tabelle si ha:

Dati		Sezione 2	
M	0.43	A2/A*	1.0004783
A1/A2	1.5	M2	0.9762034
A1/A*	1.500717501		

Soluzione delle equazioni

Trascurando il termine dF_D e supponendo y=0 il termine dN/dx diventa:

$$\frac{dN}{dx} = 4f\frac{1}{D}$$

Supponendo che il condotto abbia una sezione retta circolare si ha:

$$D = \sqrt{\frac{4}{\pi}A}$$

Da cui:

$$\frac{dN}{dx} = 4f\frac{1}{\sqrt{\frac{4}{\pi}A}} = 2f\sqrt{\frac{\pi}{A}}$$

Che codificata diventa:

```
dNdx=2*f*sqrt(pi/A(x,PosA,DimA));
```

Soluzione delle equazioni

Cambiando le condizioni iniziali il chiamante diventa:

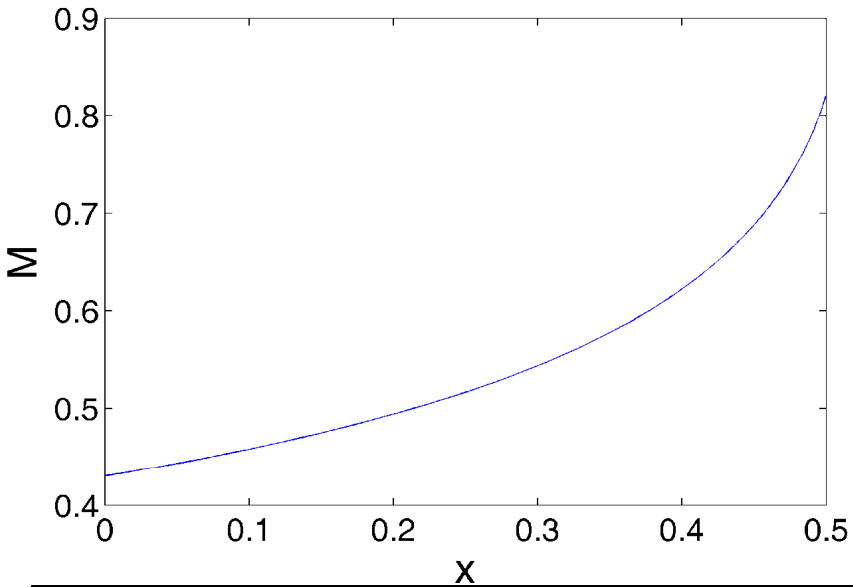
```
% \Moto1D_3\Moto1D.m
global PosA DimA gamma f;
PosA=[0;0.5;1]; % quantita' note
DimA=[1;1;3];
gamma=1.4;
f=1;
L= [0 0.5]; % dominio d'integrazione
M=.43;
M0=[M^2 DimA(1)];  %Condizione iniziale

options = odeset('RelTol',1e-8, 'AbsTol',[1e-8]);
[x, M]=ode45(@dMdx,L,M0,options);
plot(x,sqrt(M(:,1)),'-' );
xlabel('x','FontSize',20);
ylabel('M','FontSize',20);
set(gca,'FontSize',16);
fprintf('M=%g\n' ,sqrt(M(end,1)));
```

Soluzione delle equazioni

Il risultato è:

M=0.821662



Mentre utilizzando le tabelle si ha:

Solo f			
Dati		Sezione 2	
A	1	4fL*/D	0.0546971
L	0.5	M2	0.8216624
f	1		
D	1.128379167		
4fL/D	1.772453851		
M	0.43		
4fL*/D	1.827150968		

Soluzione delle equazioni

Supponiamo che i termini di adduzione di energia e di massa siano lineari:

$$T_o(x) = T_o + a_T x \qquad \dot{m}(x) = m_o + a_m x$$

Da cui:

$$\frac{dT_o}{dx} = a_T \qquad \frac{d\dot{m}}{dx} = a_m$$

$$\frac{dT_o}{T_o dx} = \frac{a_T}{T_o + a_T x} \qquad \frac{d\dot{m}}{\dot{m} dx} = \frac{a_m}{m_o + a_m x}$$

Che codificate diventano:

```
dT0dxT0=aT/(To+aT*x); %Funzione nota
dmdxm=am/(mo+am*x); %Funzione nota
```

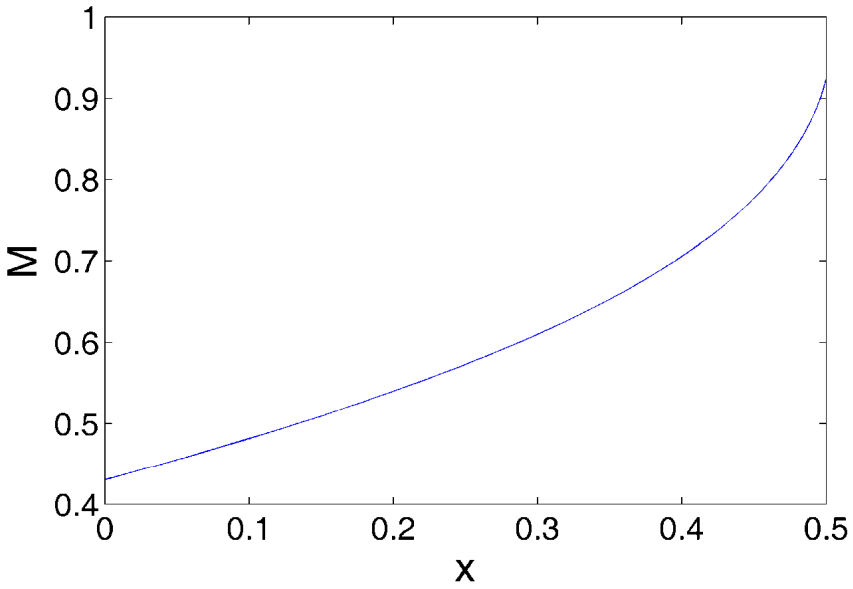

Soluzione delle equazioni

Cambiando le condizioni iniziali il chiamante diventa:

```
% \Moto1D_4\Moto1D.m
global PosA DimA gamma f To aT mo am;
% quantita' note
PosA=[0;0.5;1];
DimA=[1.5;1;3];
DimA=[1;1;3];
gamma=1.4;
f=0;
To=350;
aT=1*500;
mo=1;
am=0*0.6;
L= [0 0.5];% dominio d'integrazione
M=.43;
M0=[M^2 DimA(1)]; %Condizione iniziale
```

Soluzione delle equazioni

Il risultato è:
M=0.924149



Mentre utilizzando le tabelle si ha:

Dati		Sezione 2	
A	1	To2	600
L	0.5	To/To*	0.9955816
To	350	M2	0.9241491
aT	500		
M	0.43		
To/To*	0.58075594		

Soluzione delle equazioni

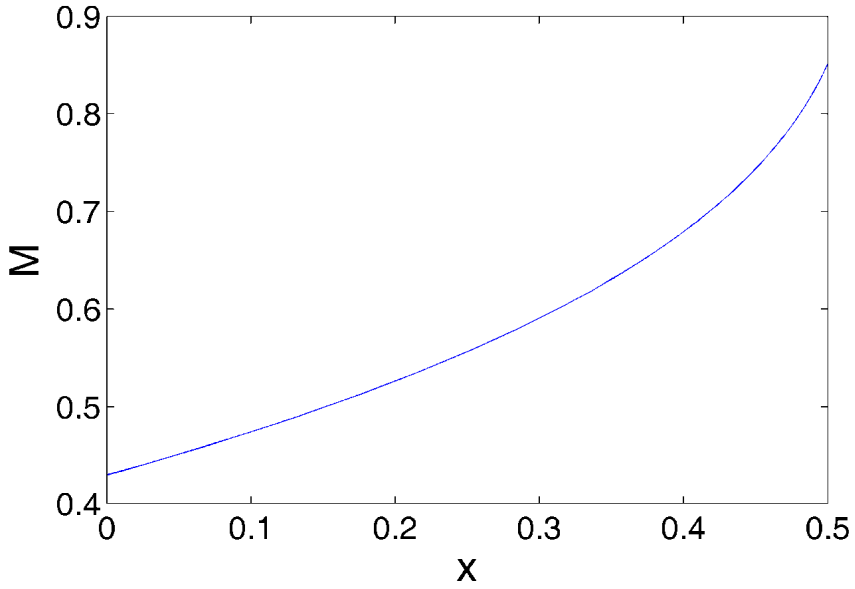
Cambiando le condizioni iniziali il chiamante diventa:

```
% \Moto1D_4\Moto1D.m
global PosA DimA gamma f To aT mo am;
% quantita' note
PosA=[0;0.5;1];
DimA=[1.5;1;3];
DimA=[1;1;3];
gamma=1.4;
f=0;
To=350;
aT=0*500;
mo=1;
am=0.6;
L= [0 0.5];% dominio d'integrazione
M=.43;
M0=[M^2 DimA(1)]; %Condizione iniziale
```

Soluzione delle equazioni

Il risultato è:

M=0.851846



Mentre utilizzando le tabelle si ha:

Dati		Sezione 2	
A	1	m2	1.3
L	0.5	m/m*	0.9906955
m	1	M2	0.8518461
am	0.6		
M	0.43		
m/m*	0.762073448		

Soluzione delle equazioni

La determinazione delle grandezze termofluidodinamiche può essere fatta utilizzando le relazioni (30-34). In particolare per la pressione statica si ha:

$$(31)\quad \frac{p_2}{p_1} = \frac{\dot{m}_2}{\dot{m}_1}\frac{A_1}{A_2}\frac{M_1}{M_2}\sqrt{\frac{T_{o2}}{T_{o1}}}\sqrt{\frac{1+\frac{\gamma-1}{2}M_1^2}{1+\frac{\gamma-1}{2}M_2^2}}$$

Che codificata diventa:

```
% \Moto1D_5\pp1.m
function pp1 = pp1(x1,M1,p1,x,M)
global PosA DimA gamma f T0 aT m0 am;
mm1=(m0+am*x)./(m0+am*x1);
ToTo1=(T0+aT*x)./(T0+aT*x1);
AA1=A(x,PosA,DimA)./A(x1,PosA,DimA);
pp1=p1*mm1.*M1.*sqrt(ToTo1.*(1+M1*M1*(gamma-1)/2)  ...
      ./((1+M.*M*(gamma-1)/2)))./(AA1.*M);
return
```

Soluzione delle equazioni

Per la Temperatura statica si ha:

$$(30)\quad \frac{T_2}{T_1} = \frac{T_{o2}}{T_{o1}} \frac{1+\frac{\gamma-1}{2}M_1^2}{1+\frac{\gamma-1}{2}M_2^2}$$

```
% \Moto1D_5\TT1.m
function TT1 = TT1(x1,M1,T1,x,M)
global PosA DimA gamma f T0 aT m0 am;
ToTo1=(T0+aT*x)./(T0+aT*x1);
TT1=T1*ToTo1.*(1+M1*M1*(gamma-1)/2)   ...
   ./((1+M.*M*(gamma-1)/2));
return
```


Soluzione delle equazioni

Si potrebbe fare un analogo discorso per le altre variabili termofluidodinamiche però se il numero di Mach, la pressione e la temperatura statiche sono noti è facile determinare sia le grandezze di ristagno (con le formule del moto isentropico) che le altre grandezze termofluidodinamiche.

```
% \Moto1D_5\pp0.m
function pp0 = pp0(M,p0)
global gamma ;
pp0=p0.*((1+M.*M*(gamma-1)/2).^(-gamma/(gamma-1)));
return

% \Moto1D_5\TT0.m
function TT0 = TT0(M,T0)
global gamma ;
TT0=T0./(1+M.*M*(gamma-1)/2);
return
```

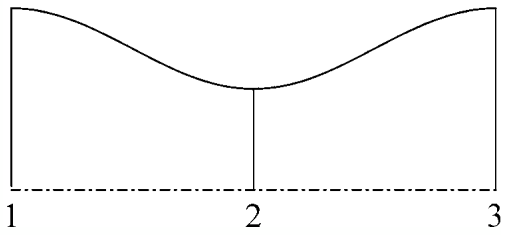
Soluzione delle equazioni

A questo punto è il caso di riordinare le varie routines in modo da poter seguire una curva di funzionamento più generale, in presenza di onda d'urto e/o sezione sonica.

A questo scopo come prima cosa riesaminiamo la funzione che descrive la variazione d'area in modo che sia possibile avere un geometria leggermente più generale.

Supponiamo che possano essere presenti più di tre sezioni principali e che la geometria sinusoidale possa essere prolungata all'esterno con un condotto a sezione costante.

$$\begin{cases} x < x_2 \quad A(x) = A_1 + (A_1 - A_2)\left(cos\left(\pi \dfrac{x - x_1}{x_2 - x_1} \right) - 1 \right) \dfrac{1}{2} \\ x < x_3 \quad A(x) = A_2 + (A_2 - A_3)\left(cos\left(\pi \dfrac{x - x_2}{x_3 - x_2} \right) - 1 \right) \dfrac{1}{2} \end{cases}$$



Soluzione delle equazioni

```
% \Moto1D_5\A.m
function A=A(x,Xs,AX)
A=zeros(size(x));
cc=find(x<=Xs(1));
A(cc)=AX(1);
for i=1:length(Xs)-1
    cc=find(x>Xs(i)  &   x<=Xs(i+1));
    A(cc)  = AX(i) + (AX(i)- AX(i+1)) * (cos(pi*(x(cc) - Xs(i)) /(Xs(i+1) -
Xs(i))) - 1) * 0.5;
end
cc=find(x>Xs(end));
A(cc)=AX(end);
```

Soluzione delle equazioni

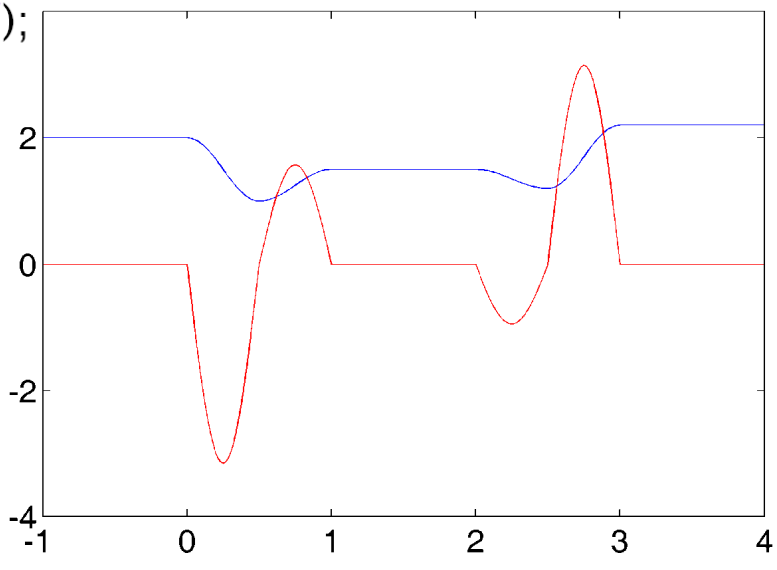
In modo analogo per la derivata:

```
% \Moto1D_5\dAdx.m
function dAdx=dAdx(x,Xs,AX)
A=zeros(size(x));
cc=find(x <= Xs(1) );
dAdx(cc)=0;
for i=1:length(Xs)-1
   cc=find(x > Xs(i)&x <= Xs(i+1) );
   dAdx(cc)  =-pi* 0.5*(AX(i)- AX(i+1)) * sin(pi*(x(cc)  - Xs(i)) /(Xs(i+1)
- Xs(i)))/(Xs(i+1) - Xs(i));
end
cc=find(x > Xs(end) );
dAdx(cc)=0;
```


Soluzione delle equazioni

Con il seguente codice chiamante si ha:

```
% \Moto1D_5\ProvaArea.m
PosA=[0;0.5;1;2;2.5;3];
DimA=[2;1;1.5;1.5;1.2;2.2];
x=-1:.01:4;
plot(x,A(x,PosA,DimA));
hold on
plot(x,dAdx(x,PosA,DimA),'r');
hold off
```



Soluzione delle equazioni

Si deve ora modificare il codice del programma principale.

Si utilizzerà la seguente simbologia:

- Le condizioni di ristagno saranno indicate con il pedice 0 (p0 =pressione di ristagno) ;
- Le sezioni saranno indicate con un numero progressivo in particolare la sezione relativa ad x=0 sarà sempre indicata con il numero 1 (p1= pressione statica nella sezione 1 relativa quindi ad x=0);
- Il moto prima della sezione 1 sarà considerato isentropico quindi, ad esempio, p0=p01;
- In presenza di discontinuità si utilizzerà il pedice aggiuntivo v o m per indicare rispettivamente le condizioni a valle ed a monte della dicontinuità;
- QM indica il numero di Mach al quadrato;
- L'assenza di pedici indica che la variabile è un vettore;

Soluzione delle equazioni

Eliminando l'area dall'equazione differenziale e riscrivendo l'equazione per il numero di Mach con le convenzioni adottate si ha:

```
function dMdx = dMdx(x,y)% \Moto1D_5\dMdx.m
global PosA DimA gamma f T0 aT m0 am;
M=y; QM=M*M; %Mach al quadrato
Appo=(1-QM)/(1+QM*(gamma-1)/2);
if Appo==0  error ( 'Divisione per zero'); end
Appo=0.5*M/Appo;
Appo1=1+QM*gamma;
dAdxA=dAdx(x,PosA,DimA)/A(x,PosA,DimA);%Funzione nota
dNdx=2*f*sqrt(pi/A(x,PosA,DimA));
dT0dxT0=aT/(T0+aT*x); %Funzione nota
dmdxm=am/(m0+am*x); %Funzione nota
dMdx=Appo*(-2*dAdxA+gamma*QM*dNdx+Appo1*dT0dxT0
+2*Appo1*dmdxm);
return
```

Soluzione delle equazioni

Il chiamante diventa:

```
% \Moto1D_5\Moto1D.m
global PosA DimA gamma f T0 aT m0 am;
% quantita' note ed inizializzazioni
M1=.43; p0=150; T0=350; x1=0; gamma=1.4;
f=0; aT=0*500; m0=1; am=0.6; PosA=[x1;0.5;1]; DimA=[1;1;3];
options = odeset('RelTol',1e-8, 'AbsTol',[1e-8]);

p1=pp0(M1,p0);T1=TT0(M1,T0);  %condizioni nella sezione 1
x2=0.5; L= [x1 x2];
[x, M]=ode45(@dMdx,L,M1,options);
M2=M(end); p2=pp1(0,M1,p1,x2,M2); T2=TT1(0,M1,T1,x2,M2);
fprintf('M2=%g p2=%g T2=%g\n'  ,M2,  p2,T2);
p=pp1(0,M1,p1,x,M);
plot(x,M,'-b' ,x,p/p0,'r');
xlabel('x','FontSize',20);
ylabel('M','FontSize',20);
set(gca,'FontSize',16);
```

Soluzione delle equazioni

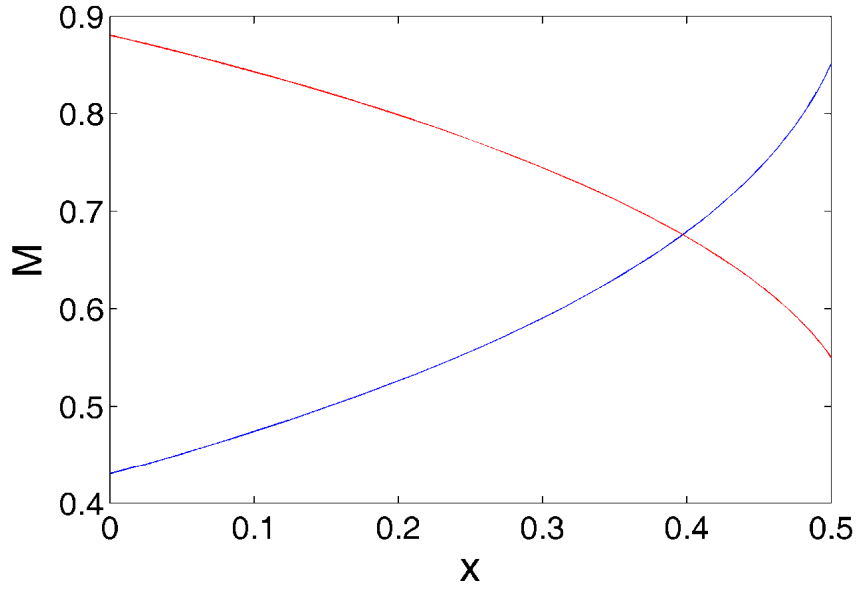
Il risultato è: M2=0.851846 p2=82.4905 T2=305.643

Mentre utilizzando le tabelle si ha:

Solo m			
Dati		Sezione 2	
A	1	m2	1.3
L	0.5	m/m*	0.990695482
po	150	M2	0.851846092
To	350	p/p*	1.190536149
m	1	T/T*	1.047917465
am	0.6	p2	82.49045649
M	0.43	T2	305.6425938
m/m*	0.762073448		
p/p*	1.906486821		
T/T*	1.157206504		
p/po	0.880650668		
T/To	0.964338753		
p	132.0976002		
T	337.5185635		

Soluzione delle equazioni

Il risultato è: M2=0.851846 p2=82.4905 T2=305.643



Soluzione delle equazioni

Il caso in cui è presente un'onda d'urto può essere facilmente affrontato integrando prima e dopo l'urto:

```
% \Moto1D_6\Moto1D.m
global PosA DimA gamma f T0 aT m0 am;
% quantita' note ed inizializzazioni
M1=2; p0=150; T0=550; m0=1; x1=0; gamma=1.4;
f=0.1; aT=0.*500; am=0.*0.6; PosA=[x1;0.5;1]; DimA=[1;1.;3];
options = odeset('RelTol',1e-8, 'AbsTol',[1e-8]);

p1=pp0(M1,p0);T1=TT0(M1,T0);  %condizioni nella sezione 1
x2=0.25; L= [x1 x2];
[x, M]=ode45(@dMdx,L,M1,options);
M2m=M(end); p2m=pp1(0,M1,p1,x2,M2m); T2m=TT1(0,M1,T1,x2,
M2m);
M2v=NSWM(M2m);p2v=NSWp(M2m,p2m);T2v=NSWT(M2m,T2m);
```

Soluzione delle equazioni

```
x3=0.5; L= [x2 x3];
[xtmp, Mtmp]=ode45(@dMdx,L,M2v,options);
x=[x; xtmp];M=[M; Mtmp];
M3=M(end); p3=pp1(x2,M2v,p2v,x3,M3); T3=TT1(x2,M2v,T2v,x3,
M3);
p=pp1(0,M1,p1,x,M);
fprintf('M3=%g p3=%g T3=%g p3-p(end)=%g\n'   ,M3,  p3,T3,p3-p
(end));

plot(x,M,'-' );
xlabel('x','FontSize',20);
ylabel('M','FontSize',20);
set(gca,'FontSize',16);
```

Soluzione delle equazioni

Dove NSWM, NSWp e NSWT sono le funzioni che determinano i rapporti caratteristici per le onde d'urto normali.

```
% \Moto1D_6\NSWM.m
function NSWM= NSWM(M)
global gamma ;
Appo=M.*M*(gamma-1)+2;
NSWM = sqrt(Appo./ (2*gamma * M.* M - (gamma - 1)));
return
```

```
% \Moto1D_6\NSWp.m
function NSWp= NSWp(M,p1)
global  gamma;
dum=2*gamma* M.* M;
NSWp=p1.* (dum - (gamma - 1)) / (gamma + 1);
return
```

Soluzione delle equazioni

Dove NSWM, NSWp e NSWT sono le funzioni che determinano i rapporti caratteristici per le onde d'urto normali.

```
% \Moto1D_6\NSWT.m
function NSWT= NSWT(M,T1)
global  gamma ;
dum = (2 * gamma / (gamma - 1)) * M.* M - 1;
dum1 = M .* M * ((2 * gamma / (gamma - 1)) + (gamma - 1) / 2);
NSWT = T1.*(1 + (gamma - 1) *0.5 * M .^ 2) .* dum ./ dum1;
return
```

Soluzione delle equazioni

Il risultato è:

M3=0.666255 p3=73.9934 T3=505.153 p3-p(end)=0

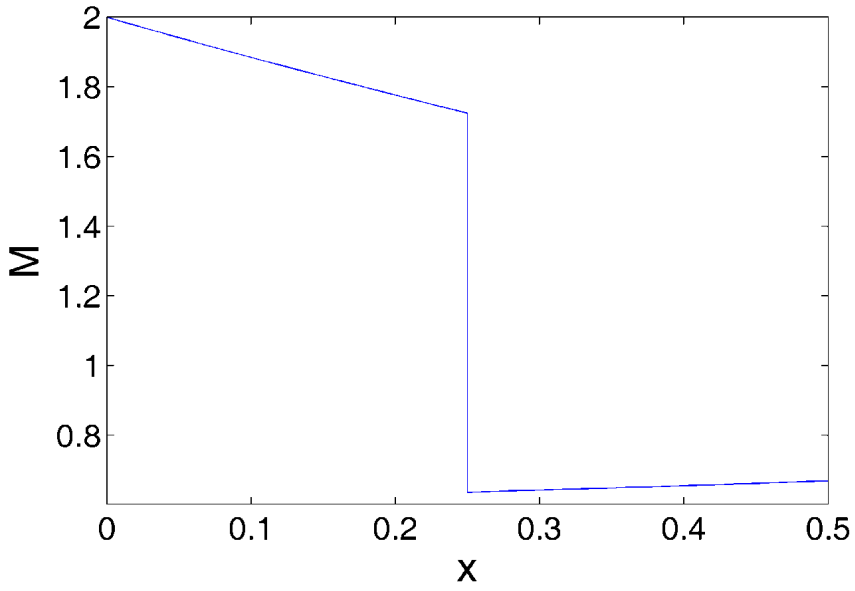
Mentre utilizzando le tabelle si ha:

Dati		Sezione 2 urto		Sezione 3	
A	1	4fL*/D	0.2163738	4fL*/D	0.282071
L	0.25	M2m	1.7247332	M3	0.666255
po	150	p/p*	0.5029171	p/p*	1.575723
To	550	T/T*	0.752379	T/T*	1.102152
f	0.1	p	23.616174	p	73.99342
D	1.128379167	T	344.84037	T	505.1529
4fL/D	0.088622693				
M	2	M2v	0.6342688		
4fL*/D	0.304996503	4fL*/D	0.3706933		
p/p*	0.40824829	p/p*	1.6615486		
T/T*	0.666666667	T/T*	1.1106387		
p/po	0.127804525	p	78.02363		
T/To	0.555555556	T	509.04273		
p	19.17067882				
T	305.5555556				

Soluzione delle equazioni

Il risultato è:

M3=0.666255 p3=73.9934 T3=505.153 p3-p(end)=0



Soluzione delle equazioni

Non è necessario valutare le grandezze termofluidodinamiche a monte ed a valle dell'onda perchè come logico sono uguali a quelle che si possono valutare direttamente.

Si esamini ora un caso più generale. Si supponga di assegnare le condizioni iniziali e la pressione ambiente e si determini la posizione dell'onda. Si può procedere con un metodo di falsa posizione.

Conviene definire una funzione Integra1D che permette di integrare l'equazione direttamente anche in presenza d'onda.

Soluzione delle equazioni

```
% \Moto1D_7\Integra1D.m
function [x, M]=Integra1D(x1,x2,xs,M1)
options = odeset('RelTol',1e-8, 'AbsTol',[1e-8]);

if (xs<x2) %urto
   L= [x1 xs];
   [x, M]=ode45(@dMdx,L,M1,options);
   M2v=NSWM(M(end));
   L= [xs x2];
   [xtmp, Mtmp]=ode45(@dMdx,L,M2v,options);
   x=[x; xtmp];M=[M; Mtmp];
else
   L= [x1 x2];
   [x, M]=ode45(@dMdx,L,M1,options);
end
return
```

Soluzione delle equazioni

Un possibile chiamante è:

```
% \Moto1D_7\Moto1D.m
global PosA DimA gamma f T0 aT m0 am;
% quantita' note ed inizializzazioni
M1=2; p0=150; T0=550; m0=1; x1=0; gamma=1.4;
f=0.1; aT=0.*500; am=0.*0.6; PosA=[x1;0.5;1]; DimA=[1;1.;3];
pa=73.99342;
p1=pp0(M1,p0);T1=TT0(M1,T0);  %condizioni nella sezione 1
x3=0.5;
x2p=0.10;

[x, M]=Integra1D(x1,x3,x2p,M1);
M3=M(end); p3=pp1(x1,M1,p1,x3,M3); T3=TT1(x1,M1,T1,x3,M3);
Err=1; Errp=p3-pa;
x2=0.12;
```

Soluzione delle equazioni

```
while (abs(Err)>10e-10)
   [x, M]=Integra1D(x1,x3,x2,M1);
   M3=M(end); p3=pp1(x1,M1,p1,x3,M3); T3=TT1(x1,M1,T1,x3,M3);
   Err=p3-pa;
   x2save=(x2p*Err-x2*Errp)/(Err-Errp);
   x2p=x2;x2=x2save;Errp=Err;
%   fprintf('%g %g \n',x2, Err);
end
fprintf('M3=%g p3=%g T3=%g p3-pa=%g\n'   ,M3,  p3,T3,p3-pa);

plot(x,M,'-' );
xlabel('x','FontSize',20);
ylabel('M','FontSize',20);
set(gca,'FontSize',16);
```

Soluzione delle equazioni

Il risultato è:

M3=0.666255 p3=73.9934 T3=505.153 p3-p(end)=0

Mentre utilizzando le tabelle si ha:

Dati		Sezione 2 urto		Sezione 3	
A	1	4fL*/D	0.2163738	4fL*/D	0.282071
L	0.25	M2	1.7247332	M3	0.666255
po	150	p/p*	0.5029171	p/p*	1.575723
To	550	T/T*	0.752379	T/T*	1.102152
f	0.1	p	23.616174	p	73.99342
D	1.128379167	T	344.84037	T	505.1529
4fL/D	0.088622693				
M	2	M3	0.6342688		
4fL*/D	0.304996503	4fL*/D	0.3706933		
p/p*	0.40824829	p/p*	1.6615486		
T/T*	0.666666667	T/T*	1.1106387		
p/po	0.127804525	p	78.02363		
T/To	0.555555556	T	509.04273		
p	19.17067882				
T	305.5555556				

Soluzione delle equazioni

Utilizzando invece una funzione standard di matlab “fzero” è necessario riscrivere la funzione da azzerare.

```
% \Moto1D_8\ErrPUrto.m
function Err=ErrPUrto(xs,x1,x3,M1,p1,pa)
[x, M]=Integra1D(x1,x3,xs,M1);
M3=M(end); p3=pp1(x1,M1,p1,x3,M3);
Err=p3-pa;
return;
```

Soluzione delle equazioni

Il chiamante diventa:

```
% \Moto1D_8\Moto1D.m
global PosA DimA gamma f T0 aT m0 am;
% quantita' note ed inizializzazioni
M1=2; p0=150; T0=550; m0=1; x1=0; gamma=1.4;
f=0.1; aT=0.*500; am=0.*0.6; PosA=[x1;0.5;1]; DimA=[1;1.;3];
pa=73.99342;
%
p1=pp0(M1,p0);T1=TT0(M1,T0);  %condizioni nella sezione 1
x3=0.5;
options = optimset('TolFun',1e-8);
xs=fzero(@ErrPUrto,0.1,options,x1,x3,M1,p1,pa );
[x, M]=Integra1D(x1,x3,xs,M1);
M3=M(end); p3=pp1(x1,M1,p1,x3,M3); T3=TT1(x1,M1,T1,x3,M3);
fprintf('M3=%g p3=%g T3=%g p3-p(end)=%g\n'   ,M3,  p3,T3,p3-pa);
```

Soluzione delle equazioni

Utilizzando un procedimento analogo si può risolvere anche il caso tutto subsonico:

```
% \Moto1D_8\ErrPSub.m
function Err=ErrPSub(M1,x1,x3,p0,pa)
[x, M]=Integra1D(x1,x3,x3,M1);
M1=M(1);p1=pp0(M1,p0);
M3=M(end); p3=pp1(x1,M1,p1,x3,M3);
Err=p3-pa;
return;
```

Soluzione delle equazioni

Il chiamante diventa:

```
% \Moto1D_8\Moto1Da.m
global PosA DimA gamma f T0 aT m0 am;
% quantita' note ed inizializzazioni
M1=0.43; p0=150; T0=350; m0=1; x1=0; gamma=1.4;
f=1; aT=0.*500; am=0.*0.6; PosA=[x1;0.5;1]; DimA=[1;1.;3];
x3=0.5; pa=66.07730029;
%
options = optimset('TolFun',1e-8);
%options = optimset('Display','iter','TolFun',1e-8);
M1=fzero(@ErrPSub,M1,options,x1,x3,p0,pa );
p1=pp0(M1,p0);T1=TT0(M1,T0);  %condizioni nella sezione 1
[x, M]=Integra1D(x1,x3,x3,M1);
M3=M(end); p3=pp1(x1,M1,p1,x3,M3); T3=TT1(x1,M1,T1,x3,M3);
fprintf('M1=%g M3=%g p3=%g T3=%g p3-p(end)=%g\n'   , M1, M3 ,  p3,
T3,p3-pa);
```

Soluzione delle equazioni

Il risultato è:

Warning: Failure at t=4.692979e-001. Unable to meet integration tolerances without reducing the step size below the smallest

value allowed (8.881784e-016) at time t.

> In ode45 at 371

In Integra1D at 14

In ErrPSub at 3

In fzero at 403

In Moto1Da at 9

M1=0.43 M3=0.821662 p3=66.0773 T3=308.363 p3-p(end)=0

Un tentativo porta ad un errore d'integrazione però il risultato finale è comunque esatto.

Soluzione delle equazioni

M1=0.43 M3=0.821662 p3=66.0773 T3=308.363 p3-p(end)=0

Usando le tabelle:

Solo f			
Dati		Sezione 2	
A	1	4fL*/D	0.054697117
L	0.5	M2	0.821662398
po	150	p/p*	1.251394783
To	350	T/T*	1.057244672
f	1	p	66.07730029
D	1.128379167	T	308.3630294
4fL/D	1.772453851		
M	0.43		
4fL*/D	1.827150968		
p/p*	2.501710073		
T/T*	1.157206504		
p/po	0.880650668		
T/To	0.964338753		
p	132.0976002		
T	337.5185635		

Soluzione delle equazioni

La gestione di una sezione sonica può essere risolta con il metodo proposto da Beans (1970). L'equazione (12) è evidentemente singolare per M=1 però il valore della derivata del numero di Mach può essere ricavato facendo il limite per M che tende ad uno:

$$(12)\quad \frac{1-M^2}{1+\frac{\gamma-1}{2}M^2}\frac{dM^2}{M^2} = -2\frac{dA}{A} + \gamma M^2 dN + \left(1+\gamma M^2\right)\frac{dT_o}{T_o} + 2\left(1+\gamma M^2\right)\frac{d\dot{m}}{\dot{m}}$$

$$\lim_{M\to 1}\frac{dM}{dx} = \frac{M}{2}\left(1+\frac{\gamma-1}{2}M^2\right)\lim_{M\to 1}\frac{-2\frac{dA}{Adx} + \gamma M^2\frac{dN}{dx} + \left(1+\gamma M^2\right)\frac{dT_o}{T_o dx} + 2\left(1+\gamma M^2\right)\frac{d\dot{m}}{\dot{m}dx}}{1-M^2}$$

L'ultimo limite può essere valutato con la regola del Hospital.

Soluzione delle equazioni

$$\lim_{M\to 1}\frac{dM}{dx} = \frac{M}{2}\left(1+\frac{\gamma-1}{2}M^2\right)\lim_{M\to 1}\frac{-2\frac{dA}{Adx} + \gamma M^2\frac{dN}{dx} + \left(1+\gamma M^2\right)\frac{dT_o}{T_o dx} + 2\left(1+\gamma M^2\right)\frac{d\dot{m}}{\dot{m}dx}}{1-M^2}$$

$$\lim_{M\to 1}\frac{dM}{dx} = \frac{M}{2}\left(1+\frac{\gamma-1}{2}M^2\right)\frac{-2\frac{d}{dx}\frac{dA}{Adx} + \gamma M^2\frac{d}{dx}\frac{dN}{dx} + \left(1+\gamma M^2\right)\left(\frac{d}{dx}\frac{dT_o}{T_o dx} + 2\frac{d}{dx}\frac{d\dot{m}}{\dot{m}dx}\right)}{-2M\frac{dM}{dx}} +$$

$$+\frac{M}{2}\left(1+\frac{\gamma-1}{2}M^2\right)\frac{2\gamma M\frac{dM}{dx}\frac{dN}{dx} + 2\gamma M\frac{dM}{dx}\frac{dT_o}{T_o dx} + 4\gamma M\frac{dM}{dx}\frac{d\dot{m}}{\dot{m}dx}}{-2M\frac{dM}{dx}}$$

Risolvendo nella derivata del numero di Mach:

$$\left(\frac{dM}{dx}\right)^2_{M\to 1} = -\frac{\left(1+\frac{\gamma-1}{2}M^2\right)}{4}\left(-2\frac{d}{dx}\frac{dA}{Adx} + \gamma M^2\frac{d}{dx}\frac{dN}{dx} + \left(1+\gamma M^2\right)\left(\frac{d}{dx}\frac{dT_o}{T_o dx} + 2\frac{d}{dx}\frac{d\dot{m}}{\dot{m}dx}\right)\right) +$$

$$-2\gamma M\frac{\left(1+\frac{\gamma-1}{2}M^2\right)}{4}\left(\frac{dN}{dx} + \frac{dT_o}{T_o dx} + 2\frac{d\dot{m}}{\dot{m}dx}\right)\left(\frac{dM}{dx}\right)_{M\to 1}$$

Soluzione delle equazioni

$$\left(\frac{dM}{dx}\right)^2_{M\to 1} = -\frac{\left(1+\frac{\gamma-1}{2}M^2\right)}{4}\left(-2\frac{d}{dx}\frac{dA}{Adx}+\gamma M^2\frac{d}{dx}\frac{dN}{dx}+\left(1+\gamma M^2\right)\left(\frac{d}{dx}\frac{dT_o}{T_o dx}+2\frac{d}{dx}\frac{d\dot{m}}{\dot{m}dx}\right)\right)+$$

$$-2\gamma M\frac{\left(1+\frac{\gamma-1}{2}M^2\right)}{4}\left(\frac{dN}{dx}+\frac{dT_o}{T_o dx}+2\frac{d\dot{m}}{\dot{m}dx}\right)\left(\frac{dM}{dx}\right)_{M\to 1}$$

Questa è un'equazione quadratica nella derivata del numero di Mach è può essere risolta esplicitamente.

Riscrivendola come: $\left(\frac{dM}{dx}\right)^2_{M\to 1}+2b\left(\frac{dM}{dx}\right)+c=0$

$$c=\frac{\left(1+\frac{\gamma-1}{2}M^2\right)}{4}\left(-2\frac{d}{dx}\frac{dA}{Adx}+\gamma M^2\frac{d}{dx}\frac{dN}{dx}+\left(1+\gamma M^2\right)\left(\frac{d}{dx}\frac{dT_o}{T_o dx}+2\frac{d}{dx}\frac{d\dot{m}}{\dot{m}dx}\right)\right)$$

$$b=\frac{\left(1+\frac{\gamma-1}{2}M^2\right)}{4}\gamma M\left(\frac{dN}{dx}+\frac{dT_o}{T_o dx}+2\frac{d\dot{m}}{\dot{m}dx}\right)$$

Soluzione delle equazioni

Si ha:

$$\left(\frac{dM}{dx}\right)_{M\to 1} = -b\pm\sqrt{b^2-c}$$

Evidentemente ci sono due possibili soluzioni una positiva ed una negativa.

Soluzione delle equazioni

La determinazione della sezione in cui il numero di Mach è unitario è immediata utilizzando la (12):

$$(12)\quad \frac{1-M^2}{1+\frac{\gamma-1}{2}M^2}\frac{dM^2}{M^2} = -2\frac{dA}{A} + \gamma M^2 dN + \left(1+\gamma M^2\right)\frac{dT_o}{T_o} + 2\left(1+\gamma M^2\right)\frac{d\dot{m}}{\dot{m}}$$

Imponendo che il termine a destra sia nullo e nel caso in cui M=1 si ha:

$$-2\frac{dA}{Adx} + \gamma\frac{dN}{dx} + \left(1+\gamma\right)\frac{dT_o}{T_o dx} + 2\left(1+\gamma\right)\frac{d\dot{m}}{\dot{m}dx} = 0$$

Che è un equazione, in generale, non lineare nella variabile x.

Soluzione delle equazioni

L'algoritmo per la soluzione di un problema generale è analogo a quello seguito nel caso di flussi semplici. Si suppone di avere una sezione critica e si trovano le curve caratteristiche. Se la pressione ambiente è maggiore di quella critica subsonica si procede per tentativi con una curva tutta subsonica. Altrimenti si verifica l'eventuale presenza e posizione dell'onda d'urto normale.

La determinazione della sezione sonica può essere effettuata con la seguente funzione:

```
% \Moto1D_9\RHS12.m
function Err= RHS12(x)
global PosA DimA gamma f T0 aT m0 am;
dAdxA=dAdx(x,PosA,DimA)/A(x,PosA,DimA); %Funzione nota
dNdx=2*f*sqrt(pi/A(x,PosA,DimA));
dT0dxT0=aT/(T0+aT*x); %Funzione nota
dmdxm=am/(m0+am*x); %Funzione nota
Err=(-2*dAdxA+gamma*dNdx+(1+gamma)*(dT0dxT0+2*dmdxm));
return
```

Soluzione delle equazioni

La gestione del punto sonico può essere inserita direttamente nella routine dMdx:

```
function dMdx = dMdx(x,y)% \Moto1D_9\dMdx.m
global PosA DimA gamma f T0 aT m0 am FlagSezSonica;
h=1e-6;
M=y(1); QM=M*M; %Mach al quadrato
if abs(1-QM)<h
  Appo=0.25*(1+QM*(gamma-1)/2);
  Appo1=1+QM*gamma;
  dAdxA=dAdx(x,PosA,DimA)/A(x,PosA,DimA);%Funzione nota
  dNdx=2*f*sqrt(pi/A(x,PosA,DimA));
  dT0dxT0=aT/(T0+aT*x); %Funzione nota
  dmdxm=am/(m0+am*x); %Funzione nota
  b=gamma*Appo*M*(dNdx+dT0dxT0+2*dmdxm);
```


Soluzione delle equazioni

```
  x1=(x+h);  %Derivata numerica
  dA2=dAdx(x1,PosA,DimA)/A(x1,PosA,DimA);%Funzione nota
  dN2=2*f*sqrt(pi/A(x1,PosA,DimA));
  dT02=aT/(T0+aT*x1); %Funzione nota
  dm2=am/(m0+am*x1); %Funzione nota
  x1=(x-h);
  dA1=dAdx(x1,PosA,DimA)/A(x1,PosA,DimA);%Funzione nota
  dN1=2*f*sqrt(pi/A(x1,PosA,DimA));
  dT01=aT/(T0+aT*x1); %Funzione nota
  dm1=am/(m0+am*x1); %Funzione nota
  c=Appo*(-2*(dA2-dA1)+gamma*QM*(dN2-dN1)+Appo1*(dT02-
dT01)+2*Appo1*(dm2-dm1))/(2*h);
  if FlagSezSonica==1
    dMdx=-b+sqrt(b*b-c);
  else
    dMdx=-b-sqrt(b*b-c);
  end
```

Soluzione delle equazioni

```
else
   Appo=(1-QM)/(1+QM*(gamma-1)/2);
   if Appo==0  error ( 'Divisione per zero'); end
   Appo=0.5*M/Appo;
   Appo1=1+QM*gamma;
   dAdxA=dAdx(x,PosA,DimA)/A(x,PosA,DimA);%Funzione nota
   dNdx=2*f*sqrt(pi/A(x,PosA,DimA));
   dT0dxT0=aT/(T0+aT*x); %Funzione nota
   dmdxm=am/(m0+am*x); %Funzione nota
   dMdx=Appo*(-2 ↙
*dAdxA+gamma*QM*dNdx+Appo1*dT0dxT0+2*Appo1*dmdxm);
end
pause (0)%per bloccare il programma
return
```


Soluzione delle equazioni

Un possibile chiamante diventa:

```
% \Moto1D_9\Moto1D.m
global PosA DimA gamma f T0 aT m0 am  FlagSezSonica;
% quantita' note ed inizializzazioni
M1=0.0605; p0=100; T0=350; m0=1; x1=0; gamma=1.4;
f=0.1; aT=0.*500; am=0.*0.6; PosA=[x1;1;2]; DimA=[2;1.;3];
x3=2; pa=51;
FlagSezSonica=1;
%
%options = optimset('Display','iter','TolFun',1e-8);
options = optimset('TolFun',1e-8);
xstar=fzero(@RHS12,(x3-x1)/2,options);
[x, M]=Integra1D(xstar,x1,x1,1);
M1=M(end); p1=pp0(M1,p0);T1=TT0(M1,T0);  %condizioni nella↙
sezione 1
MStar=M(1); pStar=pp1(x1,M1,p1,xstar,MStar); TStar=TT1(x1,M1,T1,
xstar,MStar);
```

Soluzione delle equazioni

```
xs=fzero(@ErrPUrto,(xstar+x3)/2,options,xstar,x3,MStar,pStar,pa );
xtmp=flipud(x); Mtmp=flipud(M);
[x, M]=Integra1D(xstar,x3,xs,1);
x=[ xtmp;x];M=[ Mtmp;M];
M3=M(end); p3=pp1(x1,M1,p1,x3,M3); T3=TT1(x1,M1,T1,x3,M3);
fprintf('M1=%g M3=%g p3=%g T3=%g p3-p(end)=%g\n'   , M1, M3 ,  p3,
T3,p3-pa);

plot(x,M,'-' );
%plot(x,pp1(x1,M1,p1,x,M),'-');

xlabel('x','FontSize',20);
ylabel('M','FontSize',20);
set(gca,'FontSize',16);
```

Soluzione delle equazioni

Si Esamini ora l'esempio 8.7 del testo Hodge e Koenig:

A converging–diverging nozzle with the hyperbolic diameter distribution, where $D(x)$ and x are measured in feet,

$$D(x) = \sqrt{1 + 0.25(x - 3)^2}$$

is connected to a reservoir with stagnation conditions of 100 psia and 1000 R. The nozzle exhausts into a variable back pressure arrangement. Determine the nozzle behavior with respect to back pressure if the stagnation temperature increases by 20 percent and the mass flow rate increases by 10 percent. The increase in both temperature and mass flow rate is linear with respect to x, the nozzle axis coordinate. The friction factor is 0.01.

Soluzione delle equazioni

Si Esamini ora l'esempio 8.7 del testo Hodge e Koenig:

```
% \Moto1D_10\A.m
function A=A(x,Xs,AX)
D=sqrt(1+0.25*(x-3).^2);
A=pi/4*D.*D;
return

% \Moto1D_10\dAdx.m
function dAdx=dAdx(x,Xs,AX)
D=sqrt(1+0.25*(x-3).^2);
Dx=0.25*(x-3)./D;
dAdx=pi/2*D.*Dx;
return
```

Soluzione delle equazioni

```
% \Moto1D_10\Moto1D.m
global PosA DimA gamma f T0 aT m0 am  FlagSezSonica;
% quantita' note ed inizializzazioni
M1=0.0605; p0=100; T0=1000; m0=100; x1=0; gamma=1.4;
f=0.01; aT=20; am=1; PosA=[x1;1;2]; DimA=[2;1.;3];
x3=10; pa=51;
FlagSezSonica=1;
%
%options = optimset('Display','iter','TolFun',1e-8);
options = optimset('TolFun',1e-8);
xstar=fzero(@RHS12,(x3-x1)/2,options);
fprintf('xstar=%g \n',xstar);
```

Soluzione delle equazioni

```
[x, M]=Integra1D(xstar,x1,x1,1);
M1=M(end); p1=pp0(M1,p0);T1=TT0(M1,T0);  %condizioni nella sezione 1
MStar=M(1); pStar=pp1(x1,M1,p1,xstar,MStar); TStar=TT1(x1,M1,T1, xstar,MStar);
fprintf('M1=%g p1=%g T1=%g \n' , M1, p1,T1);

xtmp=flipud(x); Mtmp=flipud(M);
[x, M]=Integra1D(xstar,x3,x3,1);
x=[ xtmp;x];M=[ Mtmp;M];
M3=M(end); p3=pp1(x1,M1,p1,x3,M3); T3=TT1(x1,M1,T1,x3,M3);
fprintf('Mr3=%g pr3=%g Tr3=%g \n' , M3,  p3,T3);
fprintf('Mr2=%g pr1=%g Tr1=%g \n' , NSWM(M3), NSWp(M3, p3),NSWT(M3,T3));
```

Soluzione delle equazioni

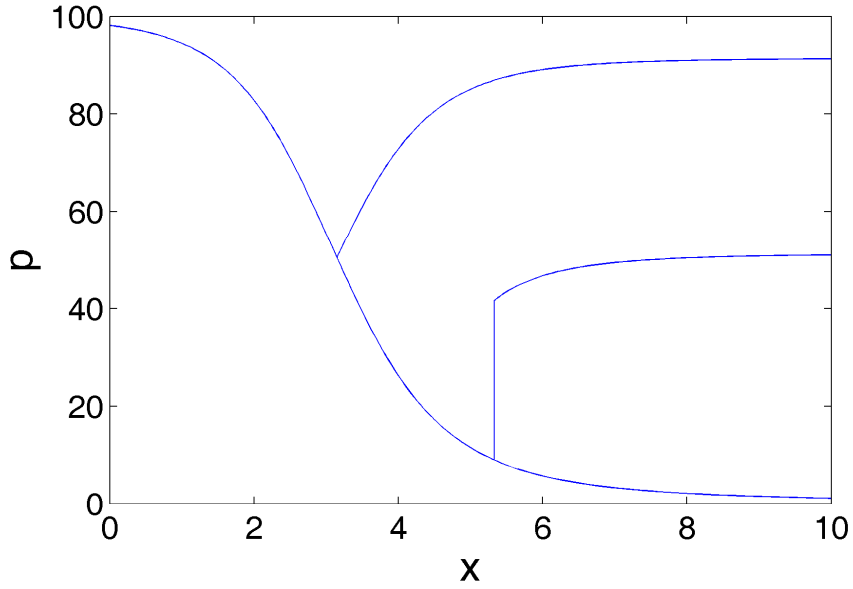
```
plot(x,pp1(x1,M1,p1,x,M),'-' );
xlabel('x','FontSize',20);
ylabel('p','FontSize',20);
set(gca,'FontSize',16);

hold on
FlagSezSonica=-1;
[x, M]=Integra1D(xstar,x3,x3,1);
M3=M(end); p3=pp1(x1,M1,p1,x3,M3); T3=TT1(x1,M1,T1,x3,M3);
fprintf('Mr1=%g pr1=%g Tr1=%g \n' , M3,  p3,T3);
plot(x,pp1(x1,M1,p1,x,M),'-' );
```

Soluzione delle equazioni

```
FlagSezSonica=1;
xs=fzero(@ErrPUrto,(xstar+x3)/2,options,xstar,x3,MStar,pStar,pa );
[x, M]=Integra1D(xstar,x3,xs,1);
M3=M(end); p3=pp1(x1,M1,p1,x3,M3); T3=TT1(x1,M1,T1,x3,M3);
fprintf('M1=%g M3=%g p3=%g T3=%g p3-p(end)=%g\n'   , M1, M3 ,  p3,
T3,p3-pa);
plot(x,pp1(x1,M1,p1,x,M),'-' );
```


Soluzione delle equazioni



I risultati sono simili a quelli del testo Hodge e Koenig:

xstar=3.14835

M1=0.163562 p1=98.1496 T1=994.678

Mr3=2.68123 pr3=1.13644 Tr3=492.248

Mr2=0.497118 pr1=9.34212 Tr1=1143.48

Mr1=0.052082 pr1=91.322 Tr1=1199.35

M1=0.163562 M3=0.0932038 p3=51 T3=1197.92 p3-p(end)=2.13163e-014

Soluzione delle equazioni

Un ugello convergente divergente sinusoidale è collegato ad un condotto rettilineo. Le sezioni caratteristiche sono:

x	A
0	2.0000
1.0000	1.0000
2.0000	3.0000
2.5000	3.5000

Supponendo che:

m_o	100.0000
p_o	2.0000
To	700.0000
aT	-100.0000
am	5.0000
f	0.0200
p_a	1.0000

Determinare le curve caratteristiche e la curva di funzionamento. Determinare inoltre la posizione della sezione critica e eventualmente la posizione dell'onda d'urto.

Soluzione delle equazioni

Soluzione:

M_{in}	0.3156
M_{out}	0.3039
p_{in}	1.8666
p_{out}	1.0000
x^*	0.9927
x_{Schock}	1.6351